

Wprowadzenie do systemu GNU Octave

Opracowanie: Jarosław Miszczak (`miszczak\(at\)iitis\(.\)gliwice\(.\)pl`)
Wersja złożona 27/05/2009

Spis treści

1	Podstawowe wiadomości	1
1.1	Instalacja	1
1.2	Dodatkowe programy	1
1.3	Rozpoczęcie pracy	1
1.4	Wbudowany system pomocy	2
1.5	Edycja poleceń	2
1.6	Kończenie pracy	2
2	Sterowanie i struktury danych	3
2.1	Liczby, tablice, zakresy...	3
2.2	Liczby	3
2.2.1	Tablice	3
2.2.2	Zakresy	4
2.2.3	Operacja na tablicach	5
2.2.4	Napisy	5
2.3	Pętle i instrukcje warunkowe	5
2.3.1	Instrukcja warunkowa if	5
2.3.2	Instrukcja switch	6
2.3.3	Pętla for	6
2.3.4	Pętla while	7
3	Kilka przykładów	8
3.1	Funkcja trygonometryczne	8
3.2	Równania kwadratowe	10
4	Formaty danych i grafika	12
4.1	Zapisywanie i wczytywanie danych	12
4.1.1	Dostępne formaty danych	12
4.1.2	Polecenia save i load	12

Streszczenie

GNU Octave jest darmowym programem przeznaczonym do obliczeń numerycznych. Składnia języka jest bardzo podobna do języka programu Matlab i Octave jest w dużym stopniu zgodny z programem Matlab.

Dokument ten opisuje podstawy użytkowania systemu obliczeń GNU Octave. Omówione są podstawowe struktury danych, składania języka programowania oraz niektóre funkcje dostępne standardowo w pakiecie.

Ten utwór objęty jest licencją

Creative Commons Attribution-NonCommercial 2.0 Poland.

Aby zobaczyć kopię niniejszej licencji przejdź na stronę

<http://creativecommons.org/licenses/by-nc/2.0/pl/>

lub napisz do

**Creative Commons,
543 Howard Street, 5th Floor,
San Francisco, California 94105, USA.**

Rozdział 1

Podstawowe wiadomości

1.1 Instalacja

Octave jest dostępny jako pakiet binarny rozprowadzany z większością dystrybucji systemu Linux¹. Dla osób pracujących w systemie Microsoft Windows istnieje wersja GNU Octave w postaci pakietu instalacyjnego – można ją znaleźć jako jeden z plików projektu OctaveForge². Alternatywą jest korzystanie z wersji skompilowanej dla platformy Cygwin³.

Dostępne są także porty dla systemów *BSD, jednakże są one czasem dość stare, zatem najprostszym rozwiązaniem jest kompilacja programu ze źródeł.

1.2 Dodatkowe programy

Do tworzenia wykresów Octave wykorzystuje program Gnuplot.

Program GNU TeXmacs⁴, dostępny dla platform Linux, *BSD, Mac OS X oraz Microsoft Windows, umożliwia łatwą integrację GNU Octave z systemem \LaTeX .

1.3 Rozpoczęcie pracy

W systemie Linux pracę z systemem GNU Octave rozpoczynamy poprzez wpisanie polecenia octave w wierszu poleceń

```
jam@pooky:~  
$ octave  
GNU Octave, version 2.9.9 (i486-pc-linux-gnu).  
Copyright (C) 2006 John W. Eaton.  
This is free software; see the source code for copying conditions.  
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or  
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.  
  
Additional information about Octave is available at http://www.octave.org.  
  
Please contribute if you find this software useful.
```

¹Jeżeli chcemy używać najnowszej wersji programu, to najlepszym rozwiązaniem jest kompilacja programu ze źródeł, ale niestety wymaga to zainstalowania sporej liczby dodatkowych bibliotek.

²<http://sourceforge.net/projects/octave/>

³<http://cygwin.com/>

⁴<http://texmacs.org/>

For more information, visit <http://www.octave.org/help-wanted.html>

Report bugs to <bug@octave.org> (but first, please read <http://www.octave.org/bugs.html> to learn how to write a helpful report).

octave:1>

W systemie Microsoft Windows instalator tworzy skrót do programu w Menu Start oraz na pulpicie.



1.4 Wbudowany system pomocy

Podczas pracy z programem GNU Octave użytkownik ma do dyspozycji wbudowany system pomocy. Przykładowo polecenie

octave:1> help trace

spowoduje wypisanie informacji na temat funkcji trace. Jeżeli dana funkcja nie jest funkcją wbudowaną (tak jak ma to miejsce w przypadku funkcji trace), to zostanie podana lokalizacja pliku z definicją tej funkcji. Natomiast samo polecenie help wyświetla spis wszystkich dostępnych operatorów i funkcji.

1.5 Edycja poleceń

Interpreter GNU Octave pozwala na łatwą edycję tekstu wpisywanego podczas sesji. Dostępna jest historia wiersza poleceń, po której można poruszać się wykorzystując klawisze  oraz .

1.6 Kończenie pracy

Zakończenie sesji GNU Octave jest możliwe poprzez wykonanie polecenia quit bądź poprzez kombinację klawiszy Control-D.

Rozdział 2

Sterowanie i struktury danych

2.1 Liczby, tablice, zakresy...

2.2 Liczby

GNU Octave pozwala na operowania na liczbach rzeczywistych i zespolonych. Dostępne są wszystkie standardowe operacje arytmetyczne, takie jak dodawanie, odejmowanie i potęgowanie. Ta ostatnia operacja może być reprezentowana jako `**` (dwie gwiazdki) lub `^` (daszek).

```
octave:1> 2**3 - 2^3
ans = 0
```

Liczby zespolone są wbudowanym typem danych. Reprezentowane są jako sumy postaci $a+bi$, gdzie i to jednostka urojona, która w GNU Octave jest reprezentowana jako jedna z liter `i`, `I`, `j` bądź `J`. Wszystkie działania określone na liczbach rzeczywistych są określone także dla liczb zespolonych. Przykładowo

```
octave:2> c1 = 2 + 3J
c1 = 2 + 3i
octave:3> c2 = 7.6 + 3.4I
c2 = 7.6000 + 3.4000i
octave:4> (c1 + c2)^2.9
ans = -161.17 + 1191.84i
```

2.2.1 Tablice

GNU Octave daje dużą swobodę jeżeli chodzi o operowanie na wektorach i macierzach. Do ich reprezentacji służy podstawowa w tym programie struktura danych czyli tablica.

Najprostszym sposobem utworzenia nowego wektora jest wymienienie wszystkich jego elementów

```
octave:5> v1 = [1,0,0]
v1 =
```

```
1  0  0
```

Otrzymujemy w ten sposób wektor wierszowy

$$v_1 = [1, 0, 0]$$

Aby otrzymać wektor kolumnowy

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

konieczne jest wykonanie operacji transponowania. Jest ona reprezentowana za pomocą apostrofu '.

```
octave:6> v1'  
ans =
```

```
1  
0  
0
```

Podobnie możemy zdefiniować nową macierz oddzielając kolejne wiersze średnikami.

```
octave:7> Id = [1,0,0;0,1,0;0,0,1]  
Id =
```

```
1  0  0  
0  1  0  
0  0  1
```

Niektóre macierze mogą być zbudowane poprzez wywołanie funkcji GNU Octave. Przykładem są macierze złożone z samych zer lub z samych jedynek. Aby uzyskać macierz o wymiarach 3 na 3 wypełnioną zerami wystarczy napisać

```
octave:8> zeros(3,4)  
ans =
```

```
0  0  0  0  
0  0  0  0  
0  0  0  0
```

Analogicznie funkcja ones wypełnia macierz o zadanych rozmiarach jedynekami.

Funkcje te są przydatne w sytuacji gdy chcemy najpierw zbudować macierz o danym kształcie, którą w kolejnych krokach wypełniamy w określony sposób.

2.2.2 Zakresy

Bardzo przydatnym mechanizmem GNU Octave, który jest zastosowany w przykładzie 3.1, są zakresy. Zakres jest wygodnym sposobem zapisania wektora wierszowego, którego elementy są równomiernie rozłożone. Do zapisu zakresu służy operator : (dwukropek).

```
octave:9> r1 = 1:10  
r1 =
```

```
1  2  3  4  5  6  7  8  9 10
```

```
octave:10> r2 = 1:2:20  
r2 =
```

```

1   3   5   7   9  11  13  15  17  19

octave:11> r1 + r2
ans =

2   5   8  11  14  17  20  23  26  29

```

W tym wypadku zakres `r1` to wektor liczb od 1 do 10, `r2` to wektor zawierający co drugą liczbę całkowitą z przedziału $[1, 20]$. Dodawanie zakresów, podobnie jak dodawanie wektorów, polega na dodawaniu element po elemencie.

2.2.3 Operacja na tablicach

Podobnie jako operacja dodawania także inne operacje arytmetyczne są zdefiniowane dla zakresów i tablic.

Jeżeli przed operatorem potęgowania (\wedge) dodamy kropkę, to potęgowanie zostanie zastosowane do każdego z elementów tablicy z osobna. Jest to prosty sposób na uzyskanie np. wszystkich sześciąt liczb z zakresu od 1 do 10

```

octave:1> x = 1:10
x =

1   2   3   4   5   6   7   8   9  10

octave:2> x.^3
ans =

1      8     27     64    125    216    343    512    729   1000

```

2.2.4 Napisy

Napisy w GNU Octave mogą być reprezentowane jako ciągi znaków ograniczone bądź cudzysłowami – "napis" – bądź apostrofami 'napis'. Ponieważ apostrof jest używany także do transponowania macierzy, bezpieczniej jest używać cudzysłówów.

2.3 Pętle i instrukcje warunkowe

Instrukcje sterujące dostępne w GNU Octave mają składnię wzorowaną na języku programowania C.

2.3.1 Instrukcja warunkowa if

Instrukcja warunkowa `if` pozwala na wykonanie bloku kodu jeżeli spełniony zostanie zadany warunek. Konstrukcja ta ma składnię przypominającą składnię języka programowania C. W poniższym przykładzie wybierany jest jeden z dwóch warunków ($x > 5$ bądź $x \leq 5$).

```

octave:1> x = pow2 (2);
octave:2> if (x > 5)
> x = x - 10;
> else

```



```
> x = x + 10;  
> endif
```

Znaki > oznaczają tu kontynuację wpisywanej komendy po naciśnięciu klawisze **ENTER**. Program sygnalizuje w ten sposób oczekiwanie na zakończenie rozpoczętej konstrukcji, takiej jak instrukcja if, bądź zamknięcie nawiasów.

Jeżeli chcemy rozróżnić pomiędzy większą ilością możliwości, to możemy zastosować konstrukcję if-elseif tak jak w poniższym przykładzie

```
octave:3> x = pow2 (4);  
octave:4> y = sin (pow2(10));  
octave:5> if (y > 0 )  
> y = pow2(sin(5))  
> elseif (y < 0 && x < 10)  
> y = cos(5)  
> endif
```

Komenda endif oznacza zamknięcie konstrukcji.

2.3.2 Instrukcja switch

Drugim sposobem warunkowego wykonywania komend jest instrukcja switch. Pozwala ona na wyszczególnienie co ma być wykonane w zależności od tego do jakiego zbioru należy zmienna.

W poniższym przykładzie zmienna text ma wartość 2, wykonane zostanie przypisanie x = 0

```
octave:29> test = 2  
test = 10  
octave:30> switch test  
> case { 1 2 3 7 8 9 }  
> x = 0  
> case { 4 5 6 }  
> x = 1  
> otherwise  
> x = -1  
> endswitch
```

2.3.3 Pętla for

Do poruszania się po tablicach służy pętla for. Do wypisania kwadratów liczb od 1 do 10 służy polecenie

```
octave:1> for i=1:10  
> i^2  
> endfor
```

znaki > oznaczają, iż program czeka na wprowadzenie komendy endfor, który oznacza zamknięcie pętli for.

2.3.4 Pętla while

Jeżeli chcemy wykonywać jakiś zestaw poleceń tylko dla danych spełniających określony warunek,

```
octave:25> i=1  
i = 1  
octave:26> while (i^2<1000)  
> i^2  
> i=i+1  
> endwhile
```

Rozdział 3

Kilka przykładów

W tym rozdziale zobaczymy jak można w praktyce wykorzystać podstawowa funkcjonalność pakietu GNU Octave. Zajmiemy się funkcjami trygonometrycznymi i rozwiązywaniem równań.

3.1 Funkcja trygonometryczne

Wykorzystując możliwości graficzne GNU Octave należy pamiętać o tym, że potrafi on posługiwać się jedynie dyskretnymi zbiorami danych.

Aby uzyskać wykres funkcji $\sin(x)$ w przedziale $[2, 2\pi]$ musimy zdefiniować odpowiednią tablicę w której przechowywane będą argumenty dla tej funkcji. W składni GNU Octave do tego celu służy operator `:`. Przykładowo

```
octave:1> t = 0:0.1:6.3;
```

wypełnia tablicę `t` liczbami od 0 do 6.3 z krokiem 0.1. Średnik kończący polecenia zapobiega wypisaniu zdefiniowanej tablicy na ekran.

Aby zobaczyć przebieg funkcji $\sin(x)$ w tym zakresie wydajemy polecenia

```
octave:2> plot (t, sin(t), "+3;sin(t);");
```

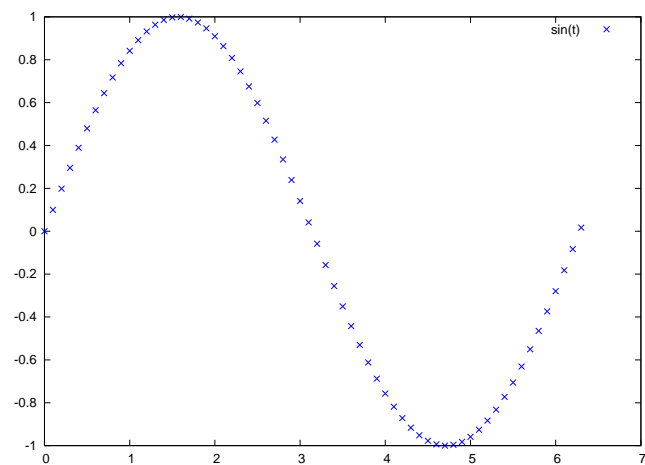
kolejne argumenty komendy `plot` to zakres na osi X, funkcja której wartości mają być wyrysowane oraz sposób sformatowania wykresu. W tym wypadku `+2;sin(t)` oznacz, iż wykres zostanie narysowany za pomocą znaków `+` w kolorze niebieskim, a do opisu wykresu zostanie wpisane `sin(t)`.

Kolory jakie udostępnia komenda `plot` to

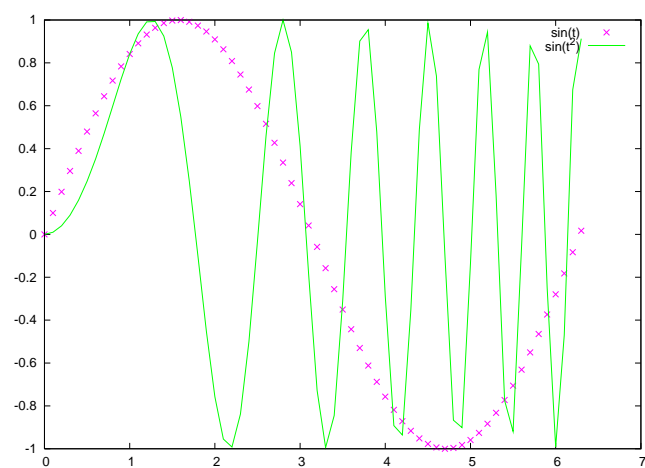
1. czerwony
2. zielony
3. niebieski
4. magenta
5. cyan
6. brązowy

Możemy też na jednym rysunku umieścić kilka wykresów funkcji.

```
octave:3> plot (t, sin(t), "+4;sin(t);", t, sin(t.^2), ";sin(t^2);");
```



Rysunek 3.1: Wykres funkcji $\sin(x)$ w przedziale $[0, 2\pi]$



Rysunek 3.2: Wykresy funkcji $\sin(x)$ oraz $\sin(x^2)$ w przedziale $[0, 2\pi]$

gdzie $t.^2$ oznacza potęgowanie zakresu element po elemencie (zobacz omówienie w sekcji 2.2.3).

Do zapisywania rysunków do pliku służy polecenie `print`¹. Polecenie to drukuje lub zapisuje do pliku wykres uzyskany przy wykonaniu ostatniej komendy `plot`. Wykres 3.2 został uzyskany poprzez wydanie polecenia

```
octave:4> print ("sin2.eps", "-deps", "-color")
```

które tworzy w bieżącym katalogu plik `sin2.eps` w formacie EPS z kolorem.

Aby uzyskać plik w formacie PNG należy wydać polecenie

```
octave:5> print ("sin2.png", "-dpng", "-color")
```

Pierwszy argument komendy `print` oznacza plik wynikowy, zaś kolejne to format wynikowy (`-dFORMAT`) i opcja powodująca wykorzystanie koloru. Inne opcje które przyjmuje ta komenda to `-portrait` bądź `-landscape` do określenia orientacji rysunku.

3.2 Równania kwadratowe

Funkcja `plot` pozwala na narysowanie wykresu dowolnej funkcji. Najprostszy przykład to funkcja kwadratowa

Równanie kwadratowe ma postać

$$a_1x^2 + a_2x + a_3 = 0, \quad (3.1)$$

gdzie współczynniki wielomianu a_1, a_2, a_3 to liczby rzeczywiste.

Współczynniki a, b, c będziemy przechowywać w trzelementowej tablicy A . Przykładowo dla $a_1 = 2, a_2 = 6$ i $a_3 = 1$ mamy

```
octave:1> A = [2, 4, 1];
```

Zdefiniujemy teraz funkcję która będzie reprezentowała nasz wielomian

```
octave:2> function y=kwadratowa(x,a)
> y = a(1)*x.^2+a(2)*x+a(3);
> endfunction
```

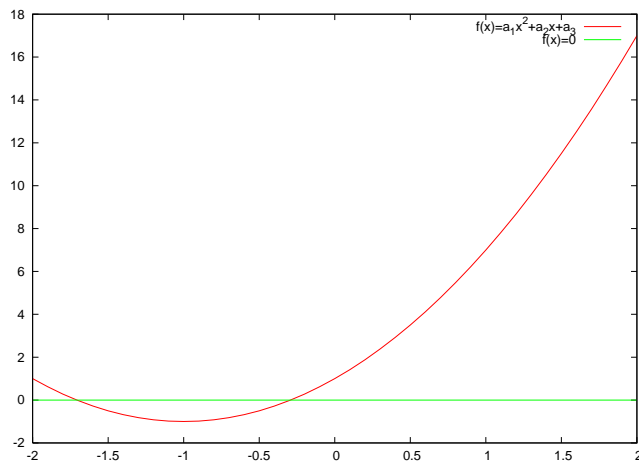
Funkcje pozwalają na zdefiniowanie nowego polecenie równoważnego grupie komend. Definicja funkcji rozpoczyna się słowem kluczowym `function`. Po nim następuje wyszczególnienie gdzie będzie przechowywana wartość zwracana – w powyższym przykładzie wartość zwracaną reprezentuje zmienna y . Dalej następuje nazwa funkcji wraz z jej argumentami. W tym wypadku mamy dwa argument – x reprezentujący argument funkcji kwadratowej oraz a reprezentująca tablicę współczynników.

Ciało funkcji zawiera komendy które zostaną wykonane podczas wywołania funkcji. Nasza funkcja zawiera jedno przypisanie które zapisuje wartość obliczeń w zmiennej y .

Warto zwrócić uwagę, że przy potęgowaniu zmienne wykorzystany został operator `.` (y = a(1)*x.^2+a(2)*x+a(3)). Pozwoli nam to przekazać tablicę jako pierwszy argument funkcji `kwadratowa`, co jest przydatne przy korzystaniu z funkcji `plot`.

Ustaliliśmy, że zmienna A przechowuje współczynniki wielomianu. Zatem

¹Te informacje dotyczą GNU Octave w wersji 2.9.x



Rysunek 3.3: Wykres funkcji $f(x) = 2x^2 + 4x + 1$ w przedziale $[-2, 2]$

```
octave:3> A
A =
```

```
    2    4    1
```

zatem aby obliczyć wartość funkcji $f(x) = 2x^2 + 4x + 1$ wystarczy napisać

```
octave:4> kwadratowa(2,A)
ans = 21
```

Natomiast aby uzyskać wykres funkcji w przedziale $[-2, 2]$ musimy zdefiniować zakres zmiennej

```
octave:5> x = -2:0.1:2;
```

a następnie wykonać polecenie

```
octave:6> plot (x, kwadratowa(x,A), ";f(x)=a_1x^2+a_2x+a_3; ")
```

Dodatkowo aby lepiej zaznaczyć punkty przecięcia z osią X, można wydać polecenie

```
octave:7> plot (x, kwadratowa(x,A), ";f(x)=a_1x^2+a_2x+a_3;",
> x, zeros(size(x)(2),1), ";f(x)=0;")
```

. Komenda `zeros(size(x)(2),1)` tworzy tablicę zer o wymiarach `size(x)(2)` na 1, czyli wektor wypełniony zerami, o długości odpowiadającej liczbie punktów przekazanych funkcji `plot`. Wynik jest przedstawiony na wykresie 3.3

Rozdział 4

Formaty danych i grafika

4.1 Zapisywanie i wczytywanie danych

4.1.1 Dostępne formaty danych

Jeżeli przeprowadzane przez nas obliczenia są czasochłonne, to dobrym pomysłem jest zapisanie wyników, tak aby nie było konieczne wielokrotne przeprowadzanie tych samych obliczeń. GNU Octave pozwala na zapis danych do kilku formatów. Podstawowym jest format tekstowy ale istnieje także możliwość zapisu w formatach wykorzystywanych przez program Matlab w wersji 4, 6 lub 7 oraz w formacie hdf5¹ (ang. *Hierarchical Data Format*). Zaletą tego ostatniego jest przenośność pomiędzy różnymi systemami obliczeń.

4.1.2 Polecenia save i load

W celu zapisania danych reprezentowanych przez zmienną `a` w pliku o nazwie `a.dat` należy wydać polecenie

```
octave:1> a = [1:4]
a =
```

```
1 2 3 4
```

```
octave:2> save a.dat a
```

Plik `a.dat` utworzony powyższym poleceniem będzie miał postać

```
# Created by Octave 2.9.3, Thu Oct 27 15:08:34 2005 CEST <miszczak@atos>
# name: a
# type: matrix
# rows: 1
# columns: 4
1 2 3 4
```

i zawiera informacje o nazwie oraz typie zapisywanego obiektu. Jeżeli plik o nazwie `a.dat` nie istniał to zostanie utworzony, natomiast jeżeli istniał, to jego dotychczasowa zawartość zostanie nadpisana.

Aby zapisać dane w formacie DHF5 konieczne jest dodanie do polecenia `save` opcji `-hd5`.

¹<http://hdf.ncsa.uiuc.edu/HDF5/>

```
octave:3> save -hdf5 a.dat a
```

Warto zwrócić uwagę, iż wielkość tak powstałego pliku jest znacznie większa – wynika to z bardziej rozbudowanej struktury wewnętrznej plików HDF5.

Bibliografia

- [1] http://www.iitis.gliwice.pl/~miszczak/projects/gnu_octave.
- [2] John W. Eaton. *GNU Octave*. <http://www.octave.org>.